

# A Bayesian Conjugate Gradient Method

Jon Cockayne<sup>\*</sup>, Chris J. Oates<sup>†</sup>, Ilse C.F. Ipsen<sup>‡</sup>, and Mark Girolami<sup>§</sup>

**Abstract.** A fundamental task in numerical computation is the solution of large linear systems. The conjugate gradient method is an iterative method which offers rapid convergence to the solution, particularly when an effective preconditioner is employed. However, for more challenging systems a substantial error can be present even after many iterations have been performed. The estimates obtained in this case are of little value unless further information can be provided about, for example, the magnitude of the error. In this paper we propose a novel statistical model for this error, set in a Bayesian framework. Our approach is a strict generalisation of the conjugate gradient method, which is recovered as the posterior mean for a particular choice of prior. The estimates obtained are analysed with Krylov subspace methods and a contraction result for the posterior is presented. The method is then analysed in a simulation study as well as being applied to a challenging problem in medical imaging.

**Keywords:** probabilistic numerics, linear systems, Krylov subspaces.

**MSC 2010 subject classifications:** 62C10, 62F15, 65F10.

## 1 Introduction

This paper presents an iterative method for solution of systems of linear equations of the form

$$A\mathbf{x}^* = \mathbf{b}, \quad (1)$$

where  $A \in \mathbb{R}^{d \times d}$  is an invertible matrix and  $\mathbf{b} \in \mathbb{R}^d$  is a vector, each given, while  $\mathbf{x}^* \in \mathbb{R}^d$  is to be determined. The principal novelty of our method, in contrast to existing approaches, is that its output is a *probability distribution* over vectors  $\mathbf{x} \in \mathbb{R}^d$  which reflects knowledge about  $\mathbf{x}^*$  after expending a limited amount of computational effort. This allows the output of the method to be used, in a principled *anytime* manner, tailored to reflect a constrained computational budget. In a special case, the mode of this distribution coincides with the estimate provided by the standard conjugate gradient method, whilst the probability mass is proven to contract onto  $\mathbf{x}^*$  as more iterations are performed.

Challenging linear systems arise in a wide variety of applications. Of these, partial differential equations (PDEs) should be emphasised, as these arise frequently throughout the applied sciences and in engineering (Evans, 2010). Finite element and finite differ-

---

<sup>\*</sup>Department of Statistics, University of Warwick, Coventry, CV4 7AL, [j.cockayne@warwick.ac.uk](mailto:j.cockayne@warwick.ac.uk), url: <http://www.joncockayne.com>

<sup>†</sup>School of Mathematics and Statistics, Herschel Building, Newcastle University, NE1 7RU, [chris.oates@ncl.ac.uk](mailto:chris.oates@ncl.ac.uk)

<sup>‡</sup>Department of Mathematics, North Carolina State University, Raleigh, NC, 27695-8205, [m.girolami@imperial.ac.uk](mailto:m.girolami@imperial.ac.uk)

<sup>§</sup>Department of Mathematics, Huxley Building, Imperial College London, London, SW7 2AZ, [ipsen@ncsu.edu](mailto:ipsen@ncsu.edu)

ence discretisations of PDEs each yield large, sparse linear systems which can sometimes be highly ill-conditioned, such as in the classically ill-posed backwards heat equation (Evans, 2010). Even for linear PDEs, a detailed discretisation may be required. This can result in a linear system with billions of degrees of freedom and require specialised algorithms to be even approximately solved practically (e.g. Reinarz et al., 2018). Another example arises in computation with Gaussian measures (Bogachev, 1998; Rasmussen, 2004), in which analytic covariance functions, such as the exponentiated quadratic, give rise to challenging linear systems. This has an impact in a number of related fields, such as symmetric collocation solution of PDEs (Fasshauer, 1999; Cockayne et al., 2016), numerical integration (Larkin, 1972; Briol et al., 2018) and generation of spatial random fields (Besag and Green, 1993; Parker and Fox, 2012; Schäfer et al., 2017). In the latter case, large linear systems must often be solved to sample from these fields, such as in models of tropical ocean surface winds (Wikle et al., 2001) where systems may again be billion-dimensional. Thus, it is clear that there exist many important situations in which error in the solution of a linear system cannot practically be avoided.

## 1.1 Linear Solvers

The solution of linear systems is one of the most ubiquitous problems in numerical analysis and Krylov subspace methods (Hestenes and Stiefel, 1952; Liesen and Strakos, 2012) are among the most successful at obtaining an approximate solution at low cost. Krylov subspace methods belong to the class of *iterative methods* (Saad, 2003), which construct a sequence  $(\mathbf{x}_m)$  that approaches  $\mathbf{x}^*$  and can be computed in an efficient manner. Iterative methods provide an alternative to *direct methods* (Davis, 2006; Allaire and Kaber, 2008) such as the LU or Cholesky decomposition, which generally incur higher cost as termination of the algorithm after  $m < d$  iterations is not meaningful. In certain cases an iterative method can produce an accurate approximation to  $\mathbf{x}^*$  with reduced computational effort and memory usage compared to a direct method.

The conjugate gradient (CG) method (Hestenes and Stiefel, 1952) is a popular iterative method, and perhaps the first instance of a Krylov subspace method. The error arising from CG can be shown to decay exponentially in the number of iterations, but convergence is slowed when the system is poorly conditioned. As a result, there is interest in solving equivalent *preconditioned* systems (Allaire and Kaber, 2008), either by solving  $P^{-1}A\mathbf{x}^* = P^{-1}\mathbf{b}$  (left-preconditioning) or  $AP^{-1}P\mathbf{x}^* = \mathbf{b}$  (right-preconditioning), where  $P$  is chosen both so that  $P^{-1}A$  (or  $AP^{-1}$ ) has a lower condition number than  $A$  itself, and so that computing the solution of systems  $P\mathbf{y} = \mathbf{c}$  is computationally inexpensive for arbitrary  $\mathbf{y}$  and  $\mathbf{c}$ . Effective preconditioning can dramatically improve convergence of CG, and of Krylov subspace methods in general, and is recommended even for well-conditioned systems owing to how rapidly conjugacy is lost in CG when implemented numerically. One reasonably generic method for sparse systems involves approximate factorisation of the matrix, through an incomplete LU or incomplete Cholesky decomposition (e.g. Ajiz and Jennings, 1984; Saad, 1994). Other common approaches exploit the structure of the problem. For example, in numerical solution of PDEs a coarse discretisation of the system can be used to construct a preconditioner for a finer discretisation (e.g. Bramble et al., 1990). A more detailed survey of preconditioning

methods can be found in many standard texts, such as Benzi (2002) and Saad (2003). However, no approach is universal, and in general careful analysis of the structure of the problem is required to determine an effective preconditioner (Saad, 2003, p. 283). At worst, constructing a good preconditioner can be as difficult as solving the linear system itself.

In situations where numerical error cannot practically be made negligible, an estimate for the error  $\mathbf{x}_m - \mathbf{x}^*$  must accompany the output  $\mathbf{x}_m$  of any linear solver. The standard approach is to analytically bound  $\|\mathbf{x}_m - \mathbf{x}^*\|$  by some function of the residual  $\|A\mathbf{x}_m - \mathbf{b}\|$ , for appropriate choices of norms, then to monitor the decay of the relative residual. In implementations, the algorithm is usually terminated when this reaches machine precision, which can require a very large number of iterations and substantial computational effort. This often constitutes the principal bottleneck in contemporary applications. The contribution of this paper is to demonstrate how Bayesian analysis can be used to develop a richer, probabilistic description for the error in estimating the solution  $\mathbf{x}^*$  with an iterative method. From a user’s perspective, this means that solutions from the presented method can still be used in a principled way, even when only a small number of iterations can be afforded.

## 1.2 Probabilistic Numerical Methods

The concept of a probabilistic numerical method dates back to Larkin (1972). The principal idea is that problems in numerical analysis can be cast as inference problems and are therefore amenable to statistical treatment. *Bayesian* probabilistic numerical methods (Cockayne et al., 2017) posit a prior distribution for the unknown, in our case  $\mathbf{x}^*$ , and condition on a finite amount of information about  $\mathbf{x}^*$  to obtain a posterior that reflects the level of uncertainty in  $\mathbf{x}^*$ , given the finite information obtained. In contemporary applications, it is common for several numerical methods to be composed in a *pipeline* to perform a complex task. For example, climate models (such as Roeckner et al., 2003) involve large systems of coupled differential equations. To simulate from these models, many approximations must be combined. Bayesian probabilistic numerical methods are of particular interest in this setting, as a probabilistic description of error can be coherently propagated through the pipeline to describe the structure of the overall error and study the contribution of each component of the pipeline to that error (Hennig et al., 2015). As many numerical methods rely on linear solvers, understanding the error incurred by these numerical methods is critical. Other works to recently highlight the value of statistical thinking in this application area includes Calvetti et al. (2018).

In recent work, Hennig (2015) treated the problem of solving (1) as an inference problem for the matrix  $A^{-1}$ , and established correspondence with existing iterative methods by selection of different matrix-valued Gaussian priors within a Bayesian framework. This approach was explored further in Bartels and Hennig (2016). There, it was observed that the posterior distribution over the matrix in Hennig (2015) produces the same factors as in the LU or Cholesky decompositions.<sup>1</sup> Our contribution takes a fun-

---

<sup>1</sup>Recall that the Cholesky decomposition is a symmetric version of the LU decomposition for symmetric positive-definite matrices.

damentally different approach, in that a prior is placed on the solution  $\mathbf{x}^*$  rather than on the matrix  $A^{-1}$ . There are advantages to the approach of Hennig (2015), in that solution of multiple systems involving the same matrix is trivial. However we argue that it is more intuitive to place a prior on  $\mathbf{x}^*$  than on  $A^{-1}$ , as one might more easily reason about the solution to a system than the elements of the inverse matrix. Furthermore, the approach of placing a prior on  $\mathbf{x}^*$  is unchanged by any left-preconditioning of the system, while the prior of Hennig (2015) is not preconditioner-invariant.

**Contribution** The main contributions of this paper are as follows:

- The *Bayesian conjugate gradient* (BayesCG) method is proposed for solution of linear systems. This is a novel probabilistic numerical method in which both prior and posterior are defined on the solution space for the linear system,  $\mathbb{R}^d$ . We argue that placing a prior on the solution space is more intuitive than existing probabilistic numerical methods and corresponds more directly with classical iterative methods. This makes substitution of BayesCG for existing iterative solvers simpler for practitioners.
- The specification of the prior distribution is discussed in detail. Several natural prior covariance structures are introduced, motivated by preconditioners or Krylov subspace methods. In addition, a hierarchical prior is proposed in which all parameters can be marginalised, allowing automatic adjustment of the posterior to the scale of the problem. This discussion provides some generic prior choices to make application of BayesCG more straightforward for users unfamiliar with probabilistic numerical methods.
- It is shown that, for a particular choice of prior, the posterior mode of BayesCG coincides with the output of the standard CG method. An explicit algorithm is provided whose complexity is shown to be a small constant factor larger than that of the standard CG method. Thus, BayesCG can be efficiently implemented and could be used in place of classical iterative methods with marginal increase in computational cost.
- A thorough convergence analysis for the new method is presented, with computational performance in mind. It is shown that the posterior mean lies in a particular Krylov subspace, and rates of convergence for the mean and contraction for the posterior are presented. The distributional quantification of uncertainty provided by this method is shown to be conservative in general.

The structure of the paper is as follows: In Section 2 BayesCG is presented and its inputs discussed. Its correspondence with CG is also established for a particular choice of prior. Section 3 demonstrates that the mean from BayesCG lies in a particular Krylov subspace and presents a convergence analysis of the method. In Section 4 the critical issue of prior choice is addressed. Several choices of prior covariance are discussed and a hierarchical prior is introduced to allow BayesCG to adapt to the scale of the problem. Section 5 contains implementation details, while in Section 6 the method is applied to

a challenging problem in medical imaging which requires repeated solution of a linear system arising from the discretisation of a PDE. The paper concludes with a discussion in Section 7. Proofs of all theoretical results are provided in the electronic supplement (Cockayne et al., 2019).

## 2 Methods

We begin in Section 2.1 by defining a Bayesian probabilistic numerical method for the linear system in (1). In Section 2.2 a correspondence to the CG method is established. In Section 2.3 we discuss a particular choice of search directions that define BayesCG. Throughout this paper, note that  $A$  is not required to be symmetric positive-definite, except for in Section 2.2.

### 2.1 Probabilistic Linear Solver

In this section we present a general probabilistic numerical method for solving (1). The approach taken is Bayesian, so that the method is defined by the choice of prior and the information on which the prior is to be conditioned. For this work, the information about  $\mathbf{x}^*$  is linear and is provided by *search directions*  $\mathbf{s}_i$ ,  $i = 1, \dots, m \ll d$ , through the matrix-vector products

$$y_i := (\mathbf{s}_i^\top A) \mathbf{x}^* = \mathbf{s}_i^\top \mathbf{b}. \quad (2)$$

The matrix-vector products on the right-hand-side are assumed to be computed without error,<sup>2</sup> which implies a likelihood model in the form of a Dirac distribution:

$$p(\mathbf{y}|\mathbf{x}) = \delta(\mathbf{y} - S_m^\top A \mathbf{x}), \quad (3)$$

where  $S_m$  denotes the matrix whose columns are  $\mathbf{s}_1, \dots, \mathbf{s}_m$ . This section assumes the search directions are given *a-priori*. The specific search directions which define BayesCG will be introduced in Section 2.3.

In general the recovery of  $\mathbf{x}^*$  from  $m < d$  pieces of information is ill-posed. The prior distribution serves to regularise the problem, in the spirit of Tikhonov (1963); Stuart (2010). Linear information is well-adapted to inference with stable distributions<sup>3</sup> such as the Gaussian or Cauchy distributions, in that the posterior distribution is available in closed-form. Optimal estimation with linear information is also well-understood (cf. Traub et al., 1988). To proceed, let  $\mathbf{x}$  be a random variable, which will be used to model epistemic uncertainty regarding the true solution  $\mathbf{x}^*$ , and endow  $\mathbf{x}$  with the prior distribution

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \mathbf{x}_0, \Sigma_0), \quad (4)$$

---

<sup>2</sup>i.e. in exact arithmetic.

<sup>3</sup>Let  $X_1$  and  $X_2$  be independent copies of a random variable  $X$ . Then  $X$  is said to be *stable* if, for any constants  $\alpha, \beta > 0$ , the random variable  $\alpha X_1 + \beta X_2$  has the same distribution as  $\gamma X + \delta$  for some constants  $\gamma > 0$  and  $\delta$ .

where  $\mathbf{x}_0$  and  $\Sigma_0$  are each assumed to be known *a-priori*, an assumption that will be relaxed in Section 4. It will be assumed throughout that  $\Sigma_0$  is a symmetric and positive-definite matrix.

Having specified the prior and the information, there exists a unique Bayesian probabilistic numerical method which outputs the conditional distribution  $p(\mathbf{x}|\mathbf{y}_m)$  (Cockayne et al., 2017) where  $\mathbf{y}_m = [y_1, \dots, y_m]^\top$  satisfies  $\mathbf{y}_m = S_m^\top A \mathbf{x}^* = S_m^\top \mathbf{b}$ . This is made clear in the following result:

**Proposition 1** (Probabilistic Linear Solver). *Let  $\Lambda_m = S_m^\top A \Sigma_0 A^\top S_m$  and  $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$ . Then the posterior distribution is given by*

$$p(\mathbf{x}|\mathbf{y}_m) = \mathcal{N}(\mathbf{x}; \mathbf{x}_m, \Sigma_m),$$

where

$$\mathbf{x}_m = \mathbf{x}_0 + \Sigma_0 A^\top S_m \Lambda_m^{-1} S_m^\top \mathbf{r}_0 \quad (5)$$

$$\Sigma_m = \Sigma_0 - \Sigma_0 A^\top S_m \Lambda_m^{-1} S_m^\top A \Sigma_0. \quad (6)$$

This provides a distribution on  $\mathbb{R}^d$  that reflects the state of knowledge given the information contained in  $\mathbf{y}_m$ . The mean,  $\mathbf{x}_m$ , could be viewed as an approximation to  $\mathbf{x}^*$  that might be provided by a numerical method. From a computational perspective, the presence of the  $m \times m$  matrix  $\Lambda_m^{-1}$  could be problematic as this implies a second linear system must be solved, albeit at a lower cost  $\mathcal{O}(m^3)$ . This could be addressed to some extent by updating  $\Lambda_m^{-1}$  iteratively using the Woodbury matrix inversion lemma, though this would not reduce the overall cost. However, as the search directions can be chosen arbitrarily, this motivates a choice which *diagonalises*  $\Lambda_m$ , to make the inverse trivial. This will be discussed further in Section 2.3.

Note that the posterior distribution is singular, in that  $\det(\Sigma_m) = 0$ . This is natural since what uncertainty remains in directions not yet explored is simply the restriction of the prior, in the measure-theoretic sense, to the subspace orthogonal to the columns of  $S_m^\top A$ . As a result, the posterior distribution is concentrated on a linear subspace of  $\mathbb{R}^d$ . Singularity of the posterior makes computing certain quantities difficult, such as posterior probabilities. Nevertheless,  $\Sigma_m$  can be decomposed using techniques such as the singular-value decomposition, so sampling from the posterior is straightforward.

For a positive-definite matrix  $M$ , define the matrix-induced inner-product of two vectors in  $\mathbb{R}^d$  by  $\langle \mathbf{x}, \mathbf{x}' \rangle_M = \mathbf{x}^\top M \mathbf{x}'$ , with associated norm  $\|\cdot\|_M$ . The following basic result establishes that the posterior covariance provides a connection to the error of  $\mathbf{x}_m$  when used as a point estimator:

**Proposition 2.**

$$\frac{\|\mathbf{x}_m - \mathbf{x}^*\|_{\Sigma_0^{-1}}}{\|\mathbf{x}_0 - \mathbf{x}^*\|_{\Sigma_0^{-1}}} \leq \sqrt{\text{tr}(\Sigma_m \Sigma_0^{-1})}.$$

Thus the right hand side provides an upper bound on the relative error of the estimator  $\mathbf{x}_m$  in the  $\Sigma_0^{-1}$ -norm. This is a weak result and tighter results for specific

search directions are provided later. In addition to bounding the error  $\mathbf{x}_m - \mathbf{x}^*$  in terms of the posterior covariance  $\Sigma_m$ , we can also compute the rate of contraction of the posterior covariance itself:

**Proposition 3.**

$$\text{tr}(\Sigma_m \Sigma_0^{-1}) = d - m.$$

The combination of Propositions 2 and 3 implies that the posterior mean  $\mathbf{x}_m$  is consistent and, since the posterior covariance characterises the width of the posterior, Proposition 3 can be viewed as a posterior contraction result. This result is intuitive; after exploring  $m$  linearly independent search directions,  $\mathbf{x}^*$  has been perfectly identified in an  $m$ -dimensional linear subspace of  $\mathbb{R}^d$ . Thus, after adjusting for the weighting of  $\mathbb{R}^d$  provided by the prior covariance  $\Sigma_0$ , it is natural that an appropriate measure of the size of the posterior should also converge at a rate that is linear.

## 2.2 Correspondence with the Conjugate Gradient Method

In this section we examine the correspondence of the posterior mean  $\mathbf{x}_m$  described in Proposition 1 with the CG method. It is frequently the case that Bayesian probabilistic numerical methods have some classical numerical method as their mean, due to the characterisation of the conditional mean of a probability distribution as the  $L_2$ -best element of the underlying space consistent with the information provided (Diaconis, 1988; Cockayne et al., 2017).

**The Conjugate Gradient Method** A large class of iterative methods for solving linear systems defined by positive-definite matrices  $A$  can be motivated by sequentially solving the following minimisation problem:

$$\mathbf{x}_m = \arg \min_{\mathbf{x} \in \mathcal{K}_m} \|\mathbf{x} - \mathbf{x}^*\|_A,$$

where  $\mathcal{K}_m$  is a sequence of  $m$ -dimensional linear subspaces of  $\mathbb{R}^d$ . It is straightforward to show that this is equivalent to:

$$\mathbf{x}_m = \arg \min_{\mathbf{x} \in \mathcal{K}_m} f(\mathbf{x}),$$

where  $f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^\top A \mathbf{x} - \mathbf{x}^\top \mathbf{b}$  is a convex quadratic functional. Let  $S_m \in \mathbb{R}^{d \times m}$  denote a matrix whose columns are arbitrary linearly independent search directions  $\mathbf{s}_1, \dots, \mathbf{s}_m$ , with  $\text{range}(S_m) = \mathcal{K}_m$ . Let  $\mathbf{x}_0$  denote an arbitrary starting point for the algorithm. Then  $\mathbf{x}_m = \mathbf{x}_0 + S_m \mathbf{c}$  for some  $\mathbf{c} \in \mathbb{R}^m$  which can be computed by solving  $\nabla f(\mathbf{x}_0 + S_m \mathbf{c}) = 0$ . This yields:

$$\mathbf{x}_m = \mathbf{x}_0 + S_m (S_m^\top A S_m)^{-1} S_m^\top (\mathbf{b} - A \mathbf{x}_0). \quad (7)$$

In CG (Hestenes and Stiefel, 1952) the search directions are constructed to simplify the inversion in (7) by imposing that the search directions are  $A$ -conjugate, that is,

$\langle \mathbf{s}_i^{\text{CG}}, \mathbf{s}_j^{\text{CG}} \rangle_A = 0$  whenever  $i \neq j$ . A set  $\{\mathbf{s}_i\}$  of  $A$ -conjugate vectors is also said to be  $A$ -orthogonal, while if the vectors additionally have  $\|\mathbf{s}_i\|_A = 1$  for each  $i$  they are said to be  $A$ -orthonormal. For simplicity of notation, we will usually work with  $A$ -orthonormal search directions, but in most implementations of CG the normalisation step can introduce stability issues and is therefore avoided.

Supposing that such a set of  $A$ -orthonormal search directions can be found, (7) simplifies to

$$\mathbf{x}_m^{\text{CG}} = \mathbf{x}_0^{\text{CG}} + S_m^{\text{CG}} (S_m^{\text{CG}})^\top (\mathbf{b} - A\mathbf{x}_0^{\text{CG}}) \quad (8)$$

which lends itself to an iterative numerical method:

$$\mathbf{x}_m^{\text{CG}} = \mathbf{x}_{m-1}^{\text{CG}} + \mathbf{s}_m^{\text{CG}} (\mathbf{s}_m^{\text{CG}})^\top (\mathbf{b} - A\mathbf{x}_{m-1}^{\text{CG}}).$$

Search directions are also constructed iteratively, motivated by gradient descent on the function  $f(\mathbf{x})$ , whose negative gradient is given by  $-\nabla f(\mathbf{x}) = \mathbf{b} - A\mathbf{x}$ . The initial un-normalised search direction  $\tilde{\mathbf{s}}_1^{\text{CG}}$  is chosen to be  $\tilde{\mathbf{s}}_1^{\text{CG}} = \mathbf{r}_0^{\text{CG}} = \mathbf{b} - A\mathbf{x}_0^{\text{CG}}$ , so that  $\mathbf{s}_1^{\text{CG}} = \tilde{\mathbf{s}}_1^{\text{CG}} / \|\tilde{\mathbf{s}}_1^{\text{CG}}\|_A$ . Letting  $\mathbf{r}_m^{\text{CG}} = \mathbf{b} - A\mathbf{x}_m^{\text{CG}}$ , subsequent search directions are given by

$$\tilde{\mathbf{s}}_m^{\text{CG}} := \mathbf{r}_{m-1}^{\text{CG}} - \langle \mathbf{s}_{m-1}^{\text{CG}}, \mathbf{r}_{m-1}^{\text{CG}} \rangle_A \mathbf{s}_{m-1}^{\text{CG}} \quad (9)$$

with  $\mathbf{s}_m^{\text{CG}} = \tilde{\mathbf{s}}_m^{\text{CG}} / \|\tilde{\mathbf{s}}_m^{\text{CG}}\|_A$ . This construction leads to search directions  $\mathbf{s}_1^{\text{CG}}, \dots, \mathbf{s}_m^{\text{CG}}$  which form an  $A$ -orthonormal set.

Equation (8) makes clear the following proposition, which shows that for a particular choice of prior the CG method is recovered as the posterior mean from Proposition 1:

**Proposition 4.** *Assume  $A$  is symmetric and positive-definite. Let  $\mathbf{x}_0 = \mathbf{0}$  and  $\Sigma_0 = A^{-1}$ . Then, taking  $S_m = S_m^{\text{CG}}$ , (5) reduces to  $\mathbf{x}_m = \mathbf{x}_m^{\text{CG}}$ .*

This result provides an intriguing perspective on the CG method, in that it represents the estimate produced by a rational Bayesian agent whose prior belief about  $\mathbf{x}^*$  is modelled by  $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, A^{-1})$ . Dependence of the prior on the inaccessible matrix inverse is in accordance with the findings in Hennig (2015, Theorem 2.4 and Lemma 3.4), in which an analogous result was presented. As observed in that paper, the appearance of  $A^{-1}$  in the prior covariance is not practically useful, as while the matrix inverse cancels in the expression for  $\mathbf{x}_m$ , it remains in the expression for  $\Sigma_m$ .

## 2.3 Search Directions

In this section the choice of search directions for the method in Proposition 1 will be discussed, initially by following an information-based complexity argument (Traub et al., 1988). For efficiency purposes, a further consideration is that  $\Lambda_m$  should be easy to invert. This naturally suggests that search directions should be chosen to be conjugate with respect to the matrix  $A\Sigma_0 A^\top$ , rather than  $A$ . Note that this approach *does not* require  $A$  to be positive-definite, as  $A\Sigma_0 A^\top$  is positive-definite for any non-singular  $A$ . Two choices of search direction will be discussed:



**Optimal Information** One choice is to formulate selection of  $S_m$  in a decision-theoretic framework, to obtain *optimal information* in the nomenclature of Cockayne et al. (2017). Abstractly, denote the probabilistic numerical method discussed above by  $P[\cdot; \mu, S_m] : \mathbb{R}^d \rightarrow \mathcal{P}(\mathbb{R}^d)$ , where  $\mathcal{P}(\mathbb{R}^d)$  is the set of all distributions on  $\mathbb{R}^d$ . The function  $P[\mathbf{b}; \mu, S_m]$  takes a right-hand-side  $\mathbf{b} \in \mathbb{R}^d$ , together with a prior  $\mu \in \mathcal{P}(\mathbb{R}^d)$  and a set of search directions  $S_m$  and outputs the posterior distribution from Proposition 1. Thus  $P[\mathbf{b}; \mu, S_m]$  is a measure and  $P[\mathbf{b}; \mu, S_m](d\mathbf{x})$  denotes its infinitesimal element.

For general  $\mu \in \mathcal{P}(\mathbb{R}^d)$ , define the *average risk* associated with the search directions  $S_m$  to be

$$R(S_m, \mu) = \iint L(\mathbf{x}, \mathbf{x}^*) P[A\mathbf{x}^*; \mu, S_m](d\mathbf{x}) \mu(d\mathbf{x}^*), \quad (10)$$

where  $L(\mathbf{x}, \mathbf{x}^*)$  represents a loss incurred when  $\mathbf{x}$  is used to estimate  $\mathbf{x}^*$ . This can be thought of as a measure of the performance of the probabilistic numerical method, averaged both over the class of problems described by  $\mu$  and over the output of the method. *Optimal information* in this paper concerns selection of  $S_m$  to minimise  $R(S_m, \mu)$ . The following proposition characterises optimal information for the posterior in Proposition 1 in the case of a squared-error loss function and when  $\mathbf{x}_0 = \mathbf{0}$ . Let  $A^{-\top} = (A^{-1})^\top$ , and let  $M^{\frac{1}{2}}$  denote a square-root of a symmetric positive-definite matrix  $M$  with the property that  $M^{\frac{\top}{2}} M^{\frac{1}{2}} = M$ , where  $M^{\frac{\top}{2}} = (M^{\frac{1}{2}})^\top$ .

**Proposition 5.** *Suppose  $\mu = \mathcal{N}(\mathbf{0}, \Sigma_0)$  and consider the squared-error loss  $L(\mathbf{x}, \mathbf{x}^*) = \|\mathbf{x} - \mathbf{x}^*\|_M^2$  where  $M$  is an arbitrary symmetric positive-definite matrix. Optimal information for this loss is given by*

$$S_m = A^{-\top} M^{\frac{\top}{2}} \Phi_m,$$

where  $\Phi_m$  is the matrix whose columns are the  $m$  leading eigenvectors of  $M^{\frac{1}{2}} \Sigma_0 M^{\frac{\top}{2}}$ , normalised such that  $\Phi_m^\top \Phi_m = I$ .

The dependence of the optimal information on  $A^{-\top}$  is problematic except for when  $M = A^\top A$ , which corresponds to measuring the performance of the algorithm through the residual  $\|A\mathbf{x}_m - \mathbf{b}\|_2^2$ . While this removes dependence on the inverse matrix, finding the search directions in this case requires computing the eigenvectors of  $A\Sigma_0 A^\top$ , the complexity of which would dominate the cost of computing the posterior in Proposition 1.

**Conjugacy** A second, more practical method for obtaining search directions that diagonalise  $\Lambda_m$  is similar to that taken in CG. Search directions are constructed which are conjugate to the matrix  $A\Sigma_0 A^\top$  by following a similar procedure to that described in Section 2.2.

**Proposition 6** (Conjugate Search Directions  $\implies$  Iterative Method). *Assume that the search directions are  $A\Sigma_0 A^\top$ -orthonormal. Denote  $\mathbf{r}_m = \mathbf{b} - A\mathbf{x}_m$ . Then,  $\mathbf{x}_m$  in (5) simplifies to*

$$\mathbf{x}_m = \mathbf{x}_{m-1} + \Sigma_0 A^\top \mathbf{s}_m (\mathbf{s}_m^\top \mathbf{r}_{m-1})$$

while to compute  $\Sigma_m$  in (6) it suffices to store only the vectors  $\Sigma_0 A^\top \mathbf{s}_j$ , for  $j = 1, \dots, m$ .

On the surface, the form of this posterior differs slightly from that in Proposition 1, in that the data are given by  $\mathbf{s}_m^\top \mathbf{r}_{m-1}$  rather than  $\mathbf{s}_m^\top \mathbf{r}_0$ . However, when search directions are conjugate, the two expressions are equivalent:

$$\begin{aligned} \mathbf{s}_m^\top \mathbf{r}_{m-1} &= \mathbf{s}_m^\top \mathbf{b} - \mathbf{s}_m^\top A \mathbf{x}_{m-1} \\ &= \mathbf{s}_m^\top \mathbf{b} - \mathbf{s}_m^\top A \mathbf{x}_0 - \underbrace{\mathbf{s}_m^\top A \Sigma_0 A^\top S_{m-1}^\top}_{=0} \mathbf{r}_0 = \mathbf{s}_m^\top \mathbf{r}_0. \end{aligned} \quad (11)$$

Use of  $\mathbf{s}_m^\top \mathbf{r}_{m-1}$  reduces the amount of storage required compared to direct application of (5). It also helps with stability as, while search directions can be shown to be conjugate mathematically, the accumulation of numerical error from floating point precision is such that numerical conjugacy may not hold, a point discussed further in Section S4.1 of the supplement.

An approach to constructing conjugate search directions for our probabilistic linear solver is now presented, again motivated by gradient descent.

**Proposition 7** (Bayesian Conjugate Gradient Method). *Recall the definition of the residual  $\mathbf{r}_m = \mathbf{b} - A\mathbf{x}_m$ . Denote  $\tilde{\mathbf{s}}_1 = \mathbf{r}_0$  and  $\mathbf{s}_1 = \tilde{\mathbf{s}}_1 / \|\tilde{\mathbf{s}}_1\|_{A\Sigma_0 A^\top}$ . For  $m > 1$  let*

$$\tilde{\mathbf{s}}_m = \mathbf{r}_{m-1} - \langle \mathbf{s}_{m-1}, \mathbf{r}_{m-1} \rangle_{A\Sigma_0 A^\top} \mathbf{s}_{m-1}.$$

*Further, assume  $\tilde{\mathbf{s}}_m \neq \mathbf{0}$  and let  $\mathbf{s}_m = \tilde{\mathbf{s}}_m / \|\tilde{\mathbf{s}}_m\|_{A\Sigma_0 A^\top}$ . Then for each  $m$ , the set  $\{\mathbf{s}_i\}_{i=1}^m$  is  $A\Sigma_0 A^\top$ -orthonormal, and as a result  $\Lambda_m = I$ .*

This is termed a Bayesian *conjugate gradient* method for the same reason as in CG, as search directions are chosen to be the direction of gradient descent subject to a conjugacy requirement, albeit a different one than in standard CG. In the context of Proposition 4, note that the search directions obtained coincide with those obtained from CG when  $A$  is symmetric positive-definite and  $\Sigma_0 = A^{-1}$ . Thus, BayesCG is a strict generalisation of CG. Note, however, that these search directions are constructed in a data-driven manner, in that they depend on the right-hand-side  $\mathbf{b}$ . This introduces a dependency on  $\mathbf{x}^*$  through the relationship in (1) which is not taken into account in the conditioning procedure and leads to conservative uncertainty assessment, as will be demonstrated in Section 6.1.

### 3 BayesCG as a Krylov Subspace Method

In this section a thorough theoretical analysis of the posterior will be presented. Fundamental to the analysis in this section is the concept of a *Krylov subspace*.

**Definition 8** (Krylov Subspace). *The Krylov subspace  $K_m(M, \mathbf{v})$ ,  $M \in \mathbb{R}^{d \times d}$ ,  $\mathbf{v} \in \mathbb{R}^d$  is defined as*

$$K_m(M, \mathbf{v}) := \text{span}(\mathbf{v}, M\mathbf{v}, M^2\mathbf{v}, \dots, M^m\mathbf{v}).$$

*For a vector  $\mathbf{w} \in \mathbb{R}^d$ , the shifted Krylov subspace is defined as*

$$\mathbf{w} + K_m(M, \mathbf{v}) := \text{span}(\mathbf{w} + \mathbf{v}, \mathbf{w} + M\mathbf{v}, \mathbf{w} + M^2\mathbf{v}, \dots, \mathbf{w} + M^m\mathbf{v}).$$

It is well-known that CG is a Krylov subspace method for symmetric positive-definite matrices  $A$  (Liesen and Strakos, 2012), meaning that

$$\mathbf{x}_m^{\text{CG}} = \arg \min_{\mathbf{x} \in \mathbf{x}_0 + K_{m-1}(A, \mathbf{r}_0)} \|\mathbf{x} - \mathbf{x}^*\|_A.$$

It will now be shown that the posterior mean for BayesCG, presented in Proposition 6, is a Krylov subspace method. For convenience, let  $K_m^* := \mathbf{x}_0 + K_m(\Sigma_0 A^\top A, \Sigma_0 A^\top \mathbf{r}_0)$ .

**Proposition 9.** *The BayesCG mean  $\mathbf{x}_m$  satisfies*

$$\mathbf{x}_m = \arg \min_{\mathbf{x} \in K_{m-1}^*} \|\mathbf{x} - \mathbf{x}^*\|_{\Sigma_0^{-1}}.$$

This proposition gives an alternate perspective on the observation that, when  $A$  is symmetric positive-definite and  $\Sigma_0 = A^{-1}$ , the posterior mean from BayesCG coincides with  $\mathbf{x}_m^{\text{CG}}$ : Indeed, for this choice of  $\Sigma_0$ ,  $K_m^*$  coincides with  $\mathbf{x}_0 + K_m(A, \mathbf{r}_0)$  and furthermore, since under this choice of  $\Sigma_0$  the norm minimised in Proposition 9 is  $\|\cdot\|_A$ , it is natural that the estimates  $\mathbf{x}_m$  and  $\mathbf{x}_m^{\text{CG}}$  should be identical.

Proposition 9 allows us to establish a convergence rate for the BayesCG mean which is similar to that which can be demonstrated for CG. Let  $\kappa(M) = \|M\|_2 \|M^{-1}\|_2$  denote the condition number of a matrix  $M$  in the matrix 2-norm. Now, noting that  $\kappa(\Sigma_0 A^\top A)$  is well-defined, as  $\Sigma_0$  and  $A$  are each nonsingular, we have:

**Proposition 10.**

$$\frac{\|\mathbf{x}_m - \mathbf{x}^*\|_{\Sigma_0^{-1}}}{\|\mathbf{x}_0 - \mathbf{x}^*\|_{\Sigma_0^{-1}}} \leq 2 \left( \frac{\sqrt{\kappa(\Sigma_0 A^\top A)} - 1}{\sqrt{\kappa(\Sigma_0 A^\top A)} + 1} \right)^m.$$

This rate is similar to the well-known convergence rate which for CG, in which  $\kappa(\Sigma_0 A^\top A)$  is replaced by  $\kappa(A)$ . However, since it holds that  $\kappa(A^\top A) \geq \kappa(A)$ , the convergence rate for BayesCG will often be worse than that for CG, unless  $\Sigma_0$  is chosen judiciously to reduce the condition number of  $\kappa(\Sigma_0 A^\top A)$ . Thus it appears that there is a price to be paid when uncertainty quantification is needed. This is unsurprising, as it is generally the case that uncertainty quantification is associated with additional cost over methods for which uncertainty quantification is not provided.

Nevertheless, the rate of convergence in Proposition 10 is significantly faster than the rate obtained in Proposition 2. The reason for this is that knowledge about how the search directions  $S_m$  were chosen has been exploited. The directions used in BayesCG are motivated by gradient descent on  $f(\mathbf{s})$ . Thus, if gradient descent is an effective heuristic for the problem at hand, then the magnitude of the error  $\mathbf{x}_m - \mathbf{x}^*$  will decrease at a rate which is sub-linear. The same cannot be said for  $\text{tr}(\Sigma_m \Sigma_0^{-1})$  which continues to converge linearly as proven in Proposition 3. Thus, the posterior covariance will in general be conservative when the BayesCG search directions are used. This is verified empirically in Section 6.1.

## 4 Prior Choice

The critical issue of prior choice is now examined. In Section 4.1 selection of the prior covariance structure will be discussed. Then in Section 4.2 a hierarchical prior will be introduced to address the scale of the prior.

### 4.1 Covariance Structure

When  $A$  is symmetric positive-definite, one choice which has already been discussed is to set  $\Sigma_0 = A^{-1}$ , which results in a posterior mean equal to the output of CG. However correspondance of the posterior mean with CG does not in itself justify this modelling choice from a probabilistic perspective and moreover this choice is not practical, as access to  $A^{-1}$  would give immediate access to the solution of (1). We therefore discuss some alternatives for the choice of  $\Sigma_0$ .

**Natural Prior** Taking inspiration from probabilistic numerical methods for PDEs (Cockayne et al., 2016; Owhadi, 2015), a natural choice presents itself: The object through which information about  $\mathbf{x}^*$  is extracted is  $\mathbf{b}$ , so it is natural, and mathematically equivalent, to place a relatively uninformative prior on the elements of  $\mathbf{b}$  rather than on  $\mathbf{x}^*$  itself. If  $\mathbf{b} \sim \mathcal{N}(\mathbf{0}, I)$  then the implied prior model for  $\mathbf{x}^*$  is  $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, (A^\top A)^{-1})$ . This prior is as impractical as that which aligns the posterior mean with CG, but has the attractive property that convergence is instantaneous when the search directions from Proposition 7 are used, as shown in Section S3.1 of the supplement.

**Preconditioner Prior** For systems in which a preconditioner is available, the preconditioner can be thought of as providing an approximation to the linear operator  $A$ . Inspired by the impractical natural covariance  $(A^\top A)^{-1}$ , one approach proposed in this paper is to set  $\Sigma_0 = (P^\top P)^{-1}$ , when a preconditioner  $P$  can be found. Since by design the action of  $P^{-1}$  can be computed efficiently, so too can the action of  $\Sigma_0$ . As mentioned in Section 1.1, the availability of a good preconditioner is problem-dependent.

**Krylov Subspace Prior** The analysis presented in Section 3 suggests another potential prior, in which probability mass is distributed according to an appropriate Krylov subspace  $K_n(M, \mathbf{b})$ . Consider a distribution constructed as the linear combination

$$\mathbf{x}_K = \sum_{i=0}^n w_i M^i \mathbf{b}, \quad (12)$$

where  $\mathbf{w} := (w_0, \dots, w_n) \sim \mathcal{N}(\mathbf{0}, \Phi)$  for some positive-definite matrix  $\Phi$ . The distribution on  $\mathbf{x}_K$  induced by (12) is clearly Gaussian with mean  $\mathbf{0}$ . To determine its covariance, note that the above expression can be rewritten as  $\mathbf{x}_K = K_n \mathbf{w}$ , where  $K_n \in \mathbb{R}^{d \times (n+1)}$  is the matrix whose columns form a basis of the Krylov subspace  $K_n(M, \mathbf{b})$ , as would be given by the Lanczos or Arnoldi algorithms (Golub and Van Loan, 2013, Chapter 9).

Irrespective of choice of  $K_n$ , the covariance of  $\mathbf{x}_K$  is given by  $\mathbb{E}(\mathbf{x}_K \mathbf{x}_K^\top) = K_n \Phi K_n^\top$  so that  $\mathbf{x}_K \sim \mathcal{N}(\mathbf{0}, K_n \Phi K_n^\top)$ . One issue with this approach is that the computation of

the matrix  $K_n$  is of the same computational complexity as  $n$  iterations of BayesCG, requiring  $n$  matrix-vector products. To ensure that this cost does not dominate the procedure, it is necessary to take  $n < m \ll d$ . However, in this situation  $\mathbf{x}^* \notin K_n(\mathbf{b}, M)$ , so it is necessary to add additional probability mass on the space orthogonal to  $K_n(M, \mathbf{b})$ , to ensure that  $\mathbf{x}^*$  lies in the prior support. To this end, let  $K_n^\perp(\mathbf{b}, M) = \mathbb{R}^d \setminus K_n(\mathbf{b}, M)$ , and let  $K_n^\perp$  denote a matrix whose columns span  $K_n^\perp(\mathbf{b}, M)$ . Let  $\mathbf{x}_K^\perp = K_n^\perp \mathbf{w}^\perp$ , where  $\mathbf{w}^\perp \sim \mathcal{N}(\mathbf{0}, \varphi I)$  for a scaling parameter  $\varphi \in \mathbb{R}$ . Then, the proposed Krylov subspace prior is given by

$$\mathbf{x} (= \mathbf{x}_0 + \mathbf{x}_K + \mathbf{x}_K^\perp) \sim \mathcal{N}(\mathbf{x}_0, K_n \Phi K_n^\top + \varphi K_n^\perp (K_n^\perp)^\top).$$

The selection of the parameters of this prior, and issues related to its implementation, are discussed in Section S3.2 of the supplement.

## 4.2 Covariance Scale

For the distributional output of BayesCG to be useful it must be well-calibrated. Loosely speaking, this means that the true solution  $\mathbf{x}^*$  should typically lie in a region where most of the posterior probability mass is situated. As such, the scale of the posterior variance should have the ability to adapt and reflect the difficulty of the linear system at hand. This can be challenging, partially because the magnitude of the solution vector is *a-priori* unknown and partially because of the aforementioned fact that the dependence of  $S_m$  on  $\mathbf{x}^*$  is not accounted for in BayesCG.

In this section we propose to treat the prior scale as an additional parameter to be learned; that is we consider the prior model  $p(\mathbf{x}|\nu) = \mathcal{N}(\mathbf{x}_0, \nu \Sigma_0)$ , where  $\mathbf{x}_0, \Sigma_0$  are as before, while  $\nu \in \mathbb{R}^+$ . This can be viewed as a generalised version of the prior in (4), which is recovered when  $\nu = 1$ . In this section we consider learning  $\nu$  in a hierarchical Bayesian framework, but we note that  $\nu$  could also be heuristically calibrated. An example of such a heuristic procedure is outlined in Section S4.3 of the supplement.

The approach pursued below follows a standard approach in Bayesian linear regression (Gelman et al., 2014). More generally, one could treat the entire covariance as unknown and perform similar conjugate analysis with an inverse-Wishart prior, though this extension was not explored. Consider then endowing  $\nu$  with Jeffreys' (improper) reference prior  $p(\nu) \propto \nu^{-1}$ . The conjugacy of this prior with the Gaussian distribution is such that the posterior marginal distributions  $p(\nu|\mathbf{y}_m)$  and  $p(\mathbf{x}|\mathbf{y}_m)$  can be found analytically. For the following proposition, IG denotes an inverse-gamma distribution, while  $\text{MVT}_m$  denotes a multivariate  $t$  distribution with  $m$  degrees of freedom.

**Proposition 11** (Hierarchical BayesCG). *When  $p(\mathbf{x}|\nu)$  and  $p(\nu)$  are as above, the posterior marginal for  $\nu$  is given by*

$$p(\nu|\mathbf{y}_m) = \text{IG}\left(\frac{m}{2}, \frac{1}{2} \mathbf{r}_0^\top S_m \Lambda_m^{-1} S_m^\top \mathbf{r}_0\right)$$

while the posterior marginal for  $\mathbf{x}$  is given by

$$p(\mathbf{x}|\mathbf{y}_m) = \text{MVT}_m\left(\mathbf{x}_m, \frac{\mathbf{r}_0^\top S_m \Lambda_m^{-1} S_m^\top \mathbf{r}_0}{m} \Sigma_m\right).$$

When the search directions are  $A\Sigma_0A^\top$ -orthonormal, this simplifies to

$$p(\nu|\mathbf{y}_m) = \text{IG}\left(\frac{m}{2}, \frac{m}{2}\nu_m\right)$$

$$p(\mathbf{x}|\mathbf{y}_m) = \text{MVT}_m(\mathbf{x}_m, \nu_m\Sigma_m),$$

where  $\nu_m := \|S_m^\top \mathbf{r}_0\|_2^2/m$ .

Since  $\mathbf{r}_0$  reflects the initial error  $\mathbf{x}_0 - \mathbf{x}^*$ , the quantity  $\nu_m$  can be thought of as describing the difficulty of the problem. Thus in this approach the scale of the posterior is data-dependent.

## 5 Implementation

In this section some important details of the implementation of BayesCG are discussed.

**Computational Cost** The cost of BayesCG is a constant factor higher than the cost of CG as three, rather than one, matrix-vector multiplications are required. Thus, the overall cost is  $\mathcal{O}(md^2)$  when the search directions from Proposition 7 are used. Note that this cost assumes that  $A$  and  $\Sigma_0$  are dense matrices; in the case of sparse matrices the cost of the matrix-vector multiplications is driven by the number of nonzero entries of each matrix rather than the dimension  $d$ .

**Termination Criteria** An appealing use of the posterior distribution might be to derive a probabilistic termination criterion for BayesCG. Recall from Proposition 2 that  $\mathbf{x}_m$  approaches  $\mathbf{x}^*$  at a rate bounded by  $\sigma_m := \sqrt{\text{tr}(\Sigma_m\Sigma_0^{-1})}$ , and from Proposition 3 that  $\text{tr}(\Sigma_m\Sigma_0^{-1}) = d - m$ . To decide in practice how many iterations of BayesCG should be performed we propose a termination criterion based upon the posterior distribution from Proposition 11:

$$\sigma_m^2 := \text{tr}(\Sigma_m\Sigma_0^{-1}) \times \nu_m = (d - m)\nu_m.$$

Thus, termination when  $\sigma_m < \epsilon$ , for some tolerance  $\epsilon > 0$  that is user-specified, might be a useful criterion. However, Proposition 2 is extremely conservative, and since Proposition 10 establishes a much faster rate of convergence for  $\|\mathbf{x}_m - \mathbf{x}^*\|_{\Sigma_0^{-1}}$  in the case of BayesCG search directions, this is likely to be an overcautious stopping criterion in the case of BayesCG. Furthermore, since this involves a data-driven estimate of scale, the term  $\nu_m$  is not uniformly decreasing with  $m$ . As a result, in practise we advocate using a more traditional termination criterion based upon monitoring the residual; see Golub and Van Loan (2013, Section 11.3.8) for more detail. Further research is needed to establish whether the posterior distribution can provide a useful termination criterion.

Full pseudocode for the BayesCG method, including the termination criterion, is presented in Algorithm 1. Two algebraic simplifications have been exploited here relative to the presentation in the main text; these are described in detail in Section S2 of the supplement. A Python implementation can be found at [github.com/jcockayne/bcg](https://github.com/jcockayne/bcg).

---

**Algorithm 1** Computation of the posterior distribution described in Proposition 6. The implementation is optimised compared to that given in Proposition 6; see Supplement S2 for detail. Further note that, for clarity, all required matrix-vector multiplications have been left explicit, but for efficiency these should be calculated once-per-loop and stored.  $\Sigma_m$  can be computed from this output as  $\Sigma_m = \Sigma_0 - \Sigma_F \Sigma_F^\top$ .

---

```

1: procedure BAYESCG( $A, \mathbf{b}, \mathbf{x}_0, \Sigma_0, \epsilon, m_{\max}$ )  $\triangleright$  ( $\epsilon$  the tolerance)
2:    $\Sigma_F$  initialised to a matrix of size  $(d \times 0)$   $\triangleright$  ( $m_{\min}$  the minimum # iterations)
3:    $\mathbf{r}_0 \leftarrow \mathbf{b} - A\mathbf{x}_0$   $\triangleright$  ( $m_{\max}$  the maximum # iterations)
4:    $\tilde{\mathbf{s}}_1 \leftarrow \mathbf{r}_0$ 
5:    $\tilde{\nu}_0 \leftarrow 0$ 
6:   for  $m = 1, \dots, m_{\max}$  do
7:      $E^2 \leftarrow \tilde{\mathbf{s}}_m^\top A \Sigma_0 A^\top \tilde{\mathbf{s}}_m$ 
8:      $\alpha_m \leftarrow \frac{\mathbf{r}_{m-1}^\top \mathbf{r}_{m-1}}{E^2}$ 
9:      $\mathbf{x}_m \leftarrow \mathbf{x}_{m-1} + \alpha_m \Sigma_0 A^\top \tilde{\mathbf{s}}_m$ 
10:     $\mathbf{r}_m \leftarrow \mathbf{r}_{m-1} - A\mathbf{x}_m$ 
11:     $\Sigma_F \leftarrow [\Sigma_F, \Sigma_0 A^\top \tilde{\mathbf{s}}_m / E]$ 
12:     $\tilde{\nu}_m \leftarrow \tilde{\nu}_{m-1} + \frac{(\mathbf{r}_{m-1}^\top \mathbf{r}_{m-1})^2}{E^2}$ 
13:    if  $\|\mathbf{r}_m\|_2 < \epsilon$  then
14:      break
15:    end if
16:     $\beta_m \leftarrow \frac{\mathbf{r}_m^\top \mathbf{r}_m}{\mathbf{r}_{m-1}^\top \mathbf{r}_{m-1}}$ 
17:     $\tilde{\mathbf{s}}_{m+1} \leftarrow \mathbf{r}_m + \beta_m \tilde{\mathbf{s}}_m$ 
18:  end for
19:   $\nu_m \leftarrow \tilde{\nu}_m / m$ 
20:  return  $\mathbf{x}_m, \Sigma_F, \nu_m$ 
21: end procedure

```

---

## 6 Numerical Results

In this section two numerical studies are presented. First we present a simulation study in which theoretical results are verified. Second we present an application to electrical impedance tomography, a challenging medical imaging technique in which linear systems must be repeatedly solved.

### 6.1 Simulation Study

The first experiment in this section is a simulation study, the goals of which are to empirically examine the convergence properties of BayesCG. Additional results which compare the output of the algorithm to the probabilistic approach of Hennig (2015) are presented in Section S4.2 of the supplementary material.

For our simulation study, a matrix  $A$  was generated by randomly drawing its eigenvalues  $\lambda_1, \dots, \lambda_d$  from an exponential distribution with parameter  $\gamma$ . A sparse, symmetric-positive definite matrix with these eigenvalues was then drawn using the

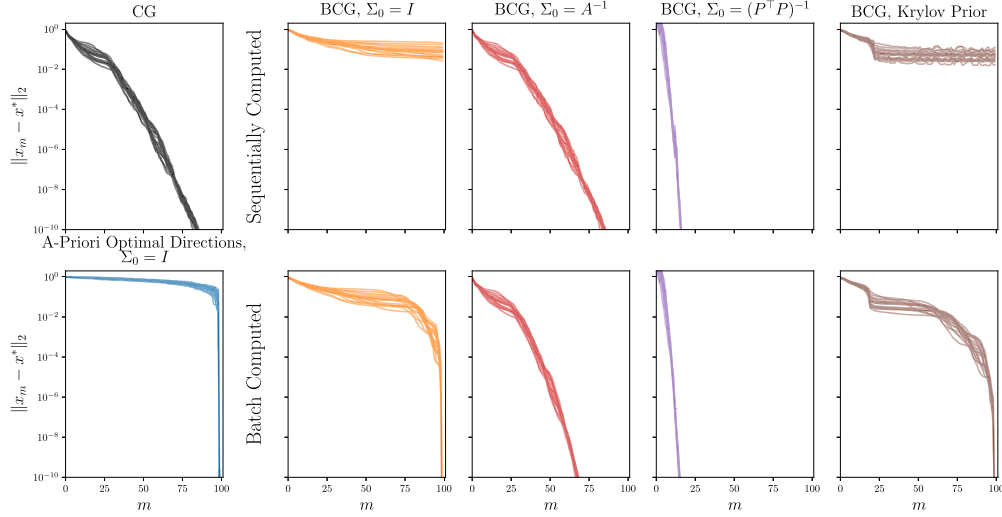


Figure 1: Convergence in mean of BayesCG (BCG). For several independent test problems,  $\mathbf{x}^* \sim \mu_{\text{ref}}$ , the error  $\|\mathbf{x}_m - \mathbf{x}^*\|_2$  was computed. The standard CG method (top left) was compared to variants of BayesCG (right), corresponding to different prior covariances  $\Sigma_0$ . The search directions used for BayesCG were either computed sequentially (top right) or in batch (bottom right). For comparison, the *a priori* optimal search directions for BayesCG are shown in the bottom left panel.

MATLAB function `sprandsym`. The proportion of non-zero entries was taken to be 20%. Subsequently, a vector  $\mathbf{x}^*$  was drawn from a reference distribution  $\mu_{\text{ref}}$  on  $\mathbb{R}^d$ , and  $\mathbf{b}$  was computed as  $\mathbf{b} = A\mathbf{x}^*$ . Throughout, the reference distribution for  $\mathbf{x}^*$  was taken to be  $\mu_{\text{ref}} = \mathcal{N}(\mathbf{0}, I)$ . For this experiment  $d = 100$  and  $\gamma = 10$ . In all cases the prior mean was taken to be  $\mathbf{x}_0 = \mathbf{0}$ . The prior covariance was alternately taken to be  $\Sigma_0 = I$ ,  $\Sigma_0 = A^{-1}$  and  $\Sigma_0 = (P^T P)^{-1}$  where  $P$  was a preconditioner found by computing an incomplete Cholesky decomposition with zero fill-in. This decomposition is simply a Cholesky decomposition in which the (approximate) factor  $\hat{L}$  has the same sparsity structure as  $A$ . The preconditioner is then given by  $P = \hat{L}\hat{L}^T$ . The matrix  $\hat{L}$  can be computed at a computational cost of  $\mathcal{O}(\text{nnz}(A)^3)$  where  $\text{nnz}(A)$  is the number of nonzero entries of  $A$ . Furthermore,  $P^{-1}$  is cheap to apply because its Cholesky factor is explicit. In addition, the Krylov subspace prior introduced in Section 4.1 has been examined. While it has been noted that the choice  $\Sigma_0 = A^{-1}$  is generally impractical, for this illustrative example  $A^{-1}$  has been computed directly. Additional experimental results which apply the methodology discussed in this section to higher-dimensional problems is presented in Section S5.

**Point Estimation** In Figure 1 the convergence of the posterior mean  $\mathbf{x}_m$  from BayesCG is contrasted with that of the output of CG, for many test problems  $\mathbf{x}^*$  with a fixed sparse matrix  $A$ . To study the impact of the numerical breakdown of conjugacy in the search directions, two choices of search directions were used; the *sequentially-computed*



search directions are those described in Proposition 7, while the *batch-computed* search directions enforce conjugacy by employing a full Gram-Schmidt orthogonalisation. The batch-computed search directions are thus given by:

$$\begin{aligned}\tilde{\mathbf{s}}_m^C &:= \mathbf{r}_{m-1} - \sum_{i=1}^{m-1} \langle \mathbf{s}_i^C, \mathbf{r}_{m-1} \rangle_{A\Sigma_0 A^\top} \mathbf{s}_i^C \\ \mathbf{s}_m^C &:= \tilde{\mathbf{s}}_m^C / \|\tilde{\mathbf{s}}_m^C\|_{A\Sigma_0 A^\top}.\end{aligned}$$

These search directions are mathematically identical to the BayesCG search directions  $\{\mathbf{s}_i\}_{i=1}^m$ , but explicitly orthogonalising with respect to all  $m-1$  previous directions ensures that numerical conjugacy is maintained. However, note that when the batch-computed search directions are used an additional loop of complexity  $\mathcal{O}(m)$  must be performed. Thus, the cost of the BayesCG algorithm with batch-computed search directions is  $\mathcal{O}(m^2 d^2)$ .

As expected from the result of Proposition 9, the convergence of the BayesCG mean vector when  $\Sigma_0 = I$  is slower than in CG. In this case, the speed of convergence for BayesCG is controlled by  $\kappa(A^\top A)$  which is larger than the corresponding  $\kappa(A)$  for CG. The *a priori* optimal search directions also appear to yield a slower rate than the BayesCG search directions, owing to the fact that they do not exploit knowledge of  $\mathbf{b}$ . Similarly as expected, the posterior mean when  $\Sigma_0 = A^{-1}$  is identical to the estimate for  $\mathbf{x}_m$  obtained from CG. The fastest rate of convergence was achieved when  $\Sigma_0 = (P^\top P)^{-1}$ , which provides a strong motivation for using a preconditioner prior if such a preconditioner can be computed, though note that a preconditioned CG method would converge at a yet faster rate gated by  $\kappa(P^{-1}A)$ .

In the lower row of Figure 1 the convergence is shown when using batch-computed directions. Here convergence appears to be faster than when using the sequentially-computed directions, at correspondingly higher computational cost. The batch-computed directions provide an exact solution after  $m = d$  iterations, in contrast to the sequentially-computed directions, for which numerical conjugacy may not hold.

Convergence for the Krylov subspace prior introduced in Section 4.1 is plotted in the right-hand column. The size of the computed subspace was set to  $n = 20$ , with  $M = A$ . The matrix  $\Phi$  was chosen to be diagonal, with  $\Phi_{ii} = [2\sigma\xi^i]^2$ , as discussed in Section S3.2 of the supplement. Here  $\sigma = \|\mathbf{x}^*\|_A$  and  $\xi = \frac{\kappa(A)-1}{\kappa(A)+1}$ , as these quantities are easily computable in this simplified setting. The remaining parameter was set to  $\gamma = 0.01$ , so that low prior weight was given to the remaining subspaces. With the sequentially computed directions significant numerical instability is observed starting at  $m = 20$ . This does not occur with the batch computed directions, where a jump in the convergence rate is seen at this iteration.

**Posterior Covariance** The full posterior output from BayesCG will now be evaluated. In Figure 2, the convergence rate of  $\text{tr}(\Sigma_m)$  is plotted for the same set of problems just described to numerically verify the result presented in Proposition 3. It is clear that when the more informative CG or BayesCG search directions are used, the rate of

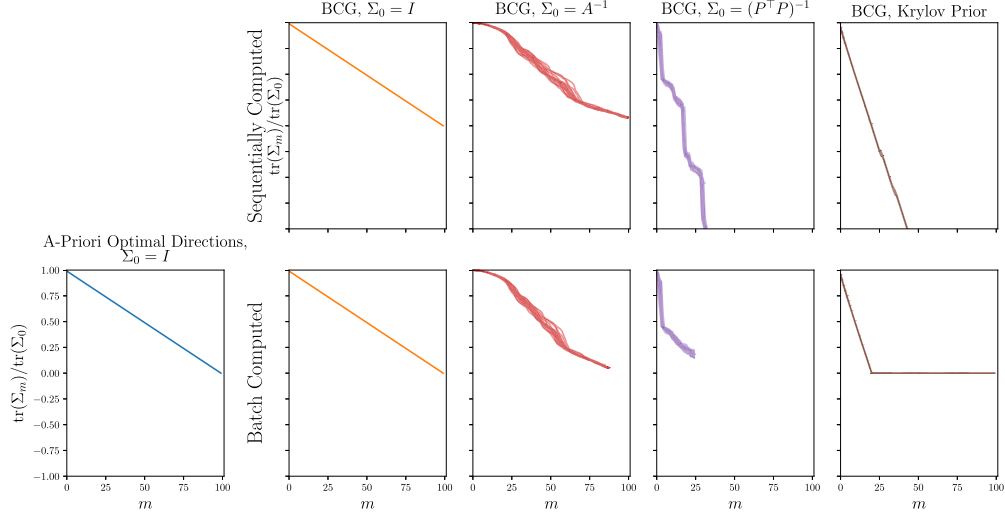


Figure 2: Convergence in posterior covariance of BayesCG (BCG), as measured by  $\text{tr}(\Sigma_m)$ . The experimental setup was as in Figure 1, here with  $\text{tr}(\Sigma_m)/\text{tr}(\Sigma_0)$  plotted.

contraction in the posterior mean does not transfer to the posterior covariance. In the remaining columns of the figure,  $\text{tr}(\Sigma_m)$  appears to contract at a roughly linear rate, in contrast to the exponential rate observed for  $\mathbf{x}_m$ . This indicates that tightening the bound provided in Proposition 3 is unlikely to be possible. Furthermore, in the last two columns of Figure 2, the impact of numerical non-conjugacy is apparent as the posterior covariance takes on negative values at around  $m = 20$ .

**Uncertainty Quantification** We now turn to an assessment of the quality of the uncertainty quantification (UQ) being provided. The same experimental setup was used as in the previous sections, however rather than running each variant of BayesCG to  $m = d$ , instead  $m = 10$  was used to ensure that UQ is needed. To avoid the issue of negative covariances seen in Figure 2, the batch-computed search directions were used throughout.

First, the Gaussian version of BayesCG from Proposition 6 was evaluated. To proceed we used the following argument: When the UQ is well-calibrated, we could consider  $\mathbf{x}^*$  as plausibly being drawn from the posterior distribution  $\mathcal{N}(\mathbf{x}_m, \Sigma_m)$ . Note that  $\Sigma_m$  is of rank  $d - m$ , but assessing uncertainty in its null space is not of interest as in this space  $\mathbf{x}^*$  has been determined exactly. Since  $\Sigma_m$  is positive semidefinite, it has the singular-value decomposition

$$\Sigma_m = U \begin{bmatrix} D & 0_{d-m,m} \\ 0_{m,d-m} & 0_{m,m} \end{bmatrix} U^\top,$$

where  $0_{m,n}$  denotes an  $m \times n$  matrix of zeroes,  $D \in \mathbb{R}^{(d-m) \times (d-m)}$  is diagonal and  $U \in \mathbb{R}^{d \times d}$  is an orthogonal matrix. The first  $d - m$  columns of  $U$ , denoted  $U_{d-m}$ , form

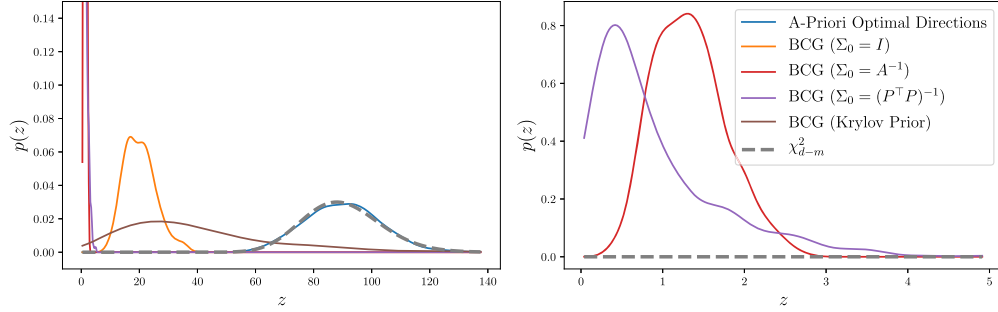


Figure 3: Assessment of the uncertainty quantification provided by the Gaussian BayesCG method, with different choices for search directions and  $\Sigma_0$ . Plotted are kernel density estimates for the statistic  $Z$  based on 500 randomly sampled test problems. These are compared with the theoretical distribution of  $Z$  when the posterior distribution is well-calibrated. The right panel zooms in on the estimate for  $\Sigma_0 = A^{-1}$  and  $\Sigma_0 = (P^T P)^{-1}$ .

a basis of  $\text{range}(\Sigma_m)$ , the subspace of  $\mathbb{R}^d$  in which  $\mathbf{x}^*$  is still uncertain. Under this hypothesis we can therefore derive a test statistic

$$U_{d-m} D^{-\frac{1}{2}} U_{d-m}^\top (\mathbf{x}^* - \mathbf{x}_m) \sim \mathcal{N}(\mathbf{0}, I_{d-m})$$

$$\implies Z(\mathbf{x}^*) := \|D^{-\frac{1}{2}} U_{d-m}^\top (\mathbf{x}^* - \mathbf{x}_m)\|_2^2 \sim \chi_{d-m}^2,$$

where here  $I_n \in \mathbb{R}^{n \times n}$  is the identity matrix. Note that the pre-factor  $U_{d-m}$  is not necessary in the final expression as  $\|\cdot\|_2$  is unitarily invariant.

Thus to evaluate the UQ we can draw many test problems  $\mathbf{x}^* \sim \mu_{\text{ref}}$ , evaluate the test statistic  $Z(\mathbf{x}^*)$  and compare the empirical distribution of this statistic to  $\chi_{d-m}^2$ . If the posterior distribution is well-calibrated we expect that the empirical distribution of the test statistic will resemble  $\chi_{d-m}^2$ . An overly-conservative posterior will exhibit a “left-shift” in its density, as  $\mathbf{x}_m$  is closer to  $\mathbf{x}^*$  than was expected. Likewise, an overly confident posterior will exhibit a “right-shift”.

In Figure 3 the empirical distribution of the statistic  $Z$  was compared to its theoretical distribution for different prior covariances. The empirical distributions were plotted as kernel density estimates based upon the computed statistic for 500 sampled test problems. Clearly the *a priori* optimal directions provide well-calibrated UQ, while for BayesCG the UQ provided by the posterior was overly-conservative for the prior covariances  $\Sigma_0 = I$ ,  $A^{-1}$  and  $(P^T P)^{-1}$ . This reflects the fact that the search directions encode knowledge of  $\mathbf{b}$ , but this knowledge is not reflected in the likelihood model used for conditioning, as discussed following Proposition 7. Furthermore, note that the quality of the UQ seems to worsen as the convergence rate for  $\mathbf{x}_m$  improves, with  $\Sigma_0 = (P^T P)^{-1}$  providing the most conservative UQ.

For the Krylov subspace prior, which encodes intuition for how search directions are selected, better UQ was provided. Though the empirical distribution of  $Z$  is not

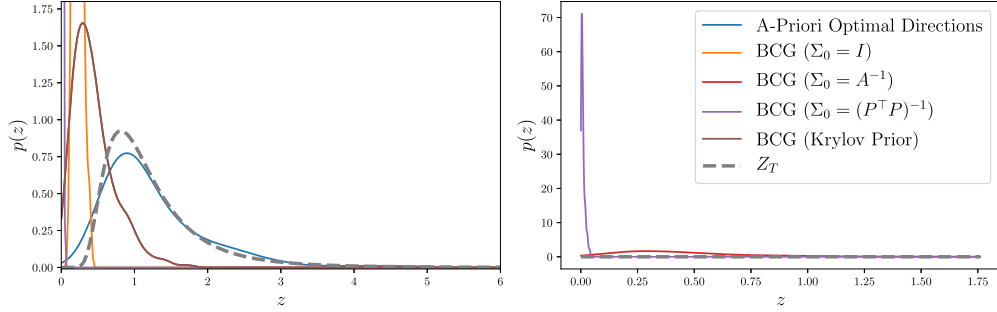


Figure 4: Assessment of the uncertainty quantification provided by the multivariate  $t$  BayesCG method, for the same prior covariances and search directions as in Figure 3.

identical to the theoretical distribution, the supports of the two distributions overlap. Thus, while the Krylov subspace prior does not fully remedy the issue caused by the use of  $\mathbf{b}$  in the search directions, some improvement is seen through the incorporation of knowledge of  $\mathbf{b}$  into the prior.

Next we assessed the UQ provided by the multivariate  $t$  posterior presented in Proposition 11. A similar procedure was followed to the Gaussian case, with a different test statistic. Let  $S \sim \mathcal{N}(\mathbf{0}, I)$ ,  $T \sim \text{MVT}_m(\boldsymbol{\mu}, \Sigma)$  and  $U \sim \chi_m^2$ . Then, it can be shown that

$$\frac{1}{\sqrt{m}} U_{d-m} D^{-\frac{1}{2}} U_{d-m}^\top (T - \boldsymbol{\mu}) \stackrel{d}{=} \frac{S}{\sqrt{U}} \implies \frac{1}{m} \|D^{-\frac{1}{2}} U_{d-m}^\top (T - \boldsymbol{\mu})\|_2^2 \stackrel{d}{=} \frac{\|S\|_2^2}{U}.$$

In the present setting,  $\boldsymbol{\mu} = \mathbf{x}_m$  and  $\Sigma = \Sigma_m$ . Furthermore  $\|S\|_2^2 \sim \chi_{d-m}^2$ . Lastly, multiplying both sides by  $m/(d-m)$  we have

$$Z(\mathbf{x}^*) := \frac{1}{d-m} \|D^{-\frac{1}{2}} U_{d-m}^\top (\mathbf{x}_m - \mathbf{x}^*)\|_2^2 \stackrel{d}{=} \frac{\frac{\|S\|_2^2}{U}}{\frac{d-m}{m}}.$$

The ratio on the right-hand-side is known to follow an  $F(d-m, m)$  distribution. In Figure 4 the empirical distribution of the test statistic  $Z(\mathbf{x}^*)$  was compared to the  $F(d-m, m)$  distribution for each of the posterior distributions considered. Again, the posterior distribution based on the *a priori* optimal search directions was well-calibrated, while the posteriors from BayesCG trade fast convergence in mean with well-calibrated UQ. As before, BayesCG with the Krylov subspace prior appears to provide the best-calibrated UQ of the (practically useful) priors considered.

Note that in both Figure 3 and Figure 4, for the choice  $\Sigma_0 = (P^\top P)^{-1}$ , which has the most rapidly converging mean in Figure 1, poor UQ properties are observed, making this otherwise appealing choice impractical. To address this we have explored a heuristic procedure for setting  $\nu_m$ , which aims to match the posterior spread to an appropriate estimate of the error  $\|\mathbf{x}_m - \mathbf{x}^*\|_2$ . This procedure is reported in Section S4 of the supplement, along with experimental results based upon it.

## 6.2 Electrical Impedance Tomography

Electrical impedance tomography (EIT) is an imaging technique used to estimate the internal conductivity of an object of interest (Somersalo et al., 1992). This conductivity is inferred from measurements of voltage induced by applying stimulating currents through electrodes attached to its boundary. EIT was originally proposed for medical applications as a non-invasive diagnostic technique (Holder, 2004), but it has also been applied in other fields, such as engineering (Oates et al., 2019).

The physical relationship between the inducing currents and resulting voltages can be described by a PDE, most commonly the complete electrode model (CEM) (Cheng et al., 1989). Consider a domain  $D \subset \mathbb{R}^n$  representing the object of interest, where typically  $n = 2$  or  $n = 3$ . Denote by  $\partial D$  the boundary of  $D$ , and let  $\sigma(\mathbf{z})$  denote the conductivity field of interest, where  $\mathbf{z} \in D$ . Denote by  $\{e_l\}_{l=1}^L$  the  $L$  electrodes, where each  $e_l \subset \partial D$  and  $e_l \cap e_m = \emptyset$  whenever  $l \neq m$ . Let  $v(\mathbf{z})$  denote the voltage field, and let  $\{I_{i,l}\}_{l=1}^L$  denote the set of stimulating currents applied to the electrodes. Let  $\{V_{i,l}^\sigma\}_{l=1}^L$  denote the corresponding voltages, and let  $\mathbf{n}$  denote the outward-pointing normal vector on  $\partial D$ . The subscript  $i$  here is to distinguish between multiple *stimulation patterns* which are generally applied in sequence and are of relevance to the inversion problem for determining  $\sigma(\mathbf{z})$  later. Denote by  $\{\zeta_l\}_{l=1}^L$  the contact impedance of each electrode. The contact impedances are used to model the fact that the contact between the electrode and the boundary of the domain is imperfect. Then the CEM is given by

$$\begin{aligned} -\nabla \cdot (\sigma(\mathbf{z}) \nabla v(\mathbf{z})) &= 0 & \mathbf{z} \in D \\ \int_{e_l} \sigma(\mathbf{z}) \frac{\partial v}{\partial \mathbf{n}}(\mathbf{z}) d\mathbf{z} &= I_{i,l} & l = 1, \dots, L \\ \sigma(\mathbf{z}) \frac{\partial v}{\partial \mathbf{n}}(\mathbf{z}) &= 0 & \mathbf{z} \in \partial D \setminus \bigcup_{l=1}^L e_l \\ v(\mathbf{z}) + \zeta_l \sigma(\mathbf{z}) \frac{\partial v}{\partial \mathbf{n}}(\mathbf{z}) &= V_{i,l}^\sigma & \mathbf{z} \in e_l, l = 1, \dots, L. \end{aligned} \quad (13)$$

A solution of this PDE is the tuple  $(v(\mathbf{z}), V_{i,1}^\sigma, \dots, V_{i,L}^\sigma)$ , consisting of the interior voltage field and the voltage measurements on the electrodes. The numerical solution of this PDE can be reduced to the solution of a linear system of the form in (1), as will shortly be explained.

Having specified the PDE linking stimulating currents to resulting voltages, it remains to describe the approach for determining  $\sigma(\mathbf{z})$  from noisy voltage measurements. These physical voltage measurements are denoted by the matrix  $V \in \mathbb{R}^{L \times (L-1)}$ , where  $V_{i,l}$  is the voltage obtained from stimulation pattern  $i$  at electrode  $l$ . The recovery problem can be cast in a Bayesian framework, as formalised in Dunlop and Stuart (2016). To this end, a prior distribution for the conductivity field is first posited and denoted  $\mu_\sigma$ . Then, the posterior distribution  $\mu_\sigma^V$  is defined through its Radon–Nikodym derivative with respect to the prior as

$$\frac{d\mu_\sigma^V}{d\mu_\sigma}(\sigma) \propto \exp(-\Phi(\sigma; V)),$$

where  $\Phi(\sigma; V)$  is known as a *potential* function and  $\exp(-\Phi(\sigma; V))$  is the likelihood. This posterior distribution is for an infinite-dimensional quantity-of-interest and is generically nonparametric, thus sampling techniques such as the preconditioned Crank–Nicolson (pCN) algorithm Cotter et al. (2013) are often employed to access it. Such algorithms require repeated evaluation of  $\Phi(\sigma; V)$  and thus the repeated solution of a PDE. Thus, there is interest in ensuring that  $\Phi(\sigma; V)$  can be computed at low cost.

**Experimental Setup** The experimental set-up is shown in Figure 5a and is due to Isaacson et al. (2004). This is described in detail in Section S.6 of the supplement. In the absence of specific data on the accuracy of the electrodes, and for convenience, the observational noise was assumed to be Gaussian with standard deviation  $\delta = 1$ . This implies a potential of the form:

$$\Phi(\sigma; V) = \sum_{i=1}^{L-1} \sum_{l=1}^L \frac{(V_{i,l} - V_{i,l}^\sigma)^2}{2\delta^2} = \frac{1}{2\delta^2} (\vec{V} - \vec{V}^\sigma)^\top (\vec{V} - \vec{V}^\sigma),$$

where  $V^\sigma$  is the matrix with  $(i, l)$ -entry  $V_{i,l}^\sigma$ . The notation  $\vec{V} \in \mathbb{R}^{L(L-1)}$  denotes the vectorisation of  $V$ , formed by concatenating columns of  $V$  into a vector as described in Section S4.2.

Apart from in pathological cases, there is no analytical solution to the CEM and thus evaluating  $\Phi(\sigma; V)$  requires an approximate solution of (13). Here a finite-element discretisation was used to solve the weak form of (13), as presented in Dunlop and Stuart (2016) and described in more detail in the supplement. This discretisation results in a sparse system of equations  $A\mathbf{x}^* = \mathbf{b}$ , where  $A$  is in this context referred to as a *stiffness matrix*. To compute  $A$  and  $\mathbf{b}$ , standard piecewise linear basis functions were used, and the computations were performed using the FEniCS finite-element package. A fine discretisation of the PDE will necessarily yield a high-dimensional linear system to be solved. We propose to use BayesCG to approximately solve the linear system, and propagate the solver uncertainty into the inverse problem associated with recovery of the conductivity field. In essence, this provides justification for small values of  $m$  to be used in the linear solver and yet ensure that the inferences for  $\sigma$  remain valid.

The Gaussian version of BayesCG was used throughout, as described in Proposition 6. Thus, assume that the output from BayesCG is  $\mathbf{x} \sim \mathcal{N}(\mathbf{x}_m, \Sigma_m)$ . The finite element approximation to the voltages  $V_{i,l}^\sigma$  is linearly related to the solution  $\mathbf{x}^*$  of the linear system, so that BayesCG implies a probability model for the voltages of the form  $\vec{V}^\sigma \sim \mathcal{N}(\vec{V}_m^\sigma, \Sigma_m^\sigma)$  for some  $\vec{V}_m^\sigma$  and  $\Sigma_m^\sigma$ ; for brevity we leave these expressions implicit. The approach proposed is to derive a new potential  $\hat{\Phi}$ , obtained by marginalising the posterior distribution output from BayesCG in the likelihood. It is straightforward to show that, for the Gaussian likelihood, this marginalisation results in the new potential

$$\hat{\Phi}(\sigma; V) = \frac{1}{2} (\vec{V} - \vec{V}_m^\sigma)^\top (\Sigma_m^\sigma + \delta^2 I)^{-1} (\vec{V} - \vec{V}_m^\sigma).$$

Thus, the new likelihood  $\exp(-\hat{\Phi}(\sigma; V))$  is still Gaussian, but with a covariance inflated by  $\Sigma_m^\sigma$ , which describes the level of accuracy in the BayesCG solver. It will be shown

that replacing  $\Phi$  with  $\hat{\Phi}$  leads to a posterior distribution  $\hat{\mu}_\sigma^V$  for the conductivity field which is appropriately to account for the accuracy of BayesCG.

Throughout this section the prior distribution over the conductivity field was taken to be a centered log-Gaussian distribution,  $\log(\sigma) \sim \mathcal{GP}(0, k)$ , with a Matérn 5/2 covariance as given by:

$$k(\mathbf{z}, \mathbf{z}') = a \left( 1 + \frac{\sqrt{5}\|\mathbf{z} - \mathbf{z}'\|_2}{\ell} + \frac{5\|\mathbf{z} - \mathbf{z}'\|_2^2}{3\ell^2} \right) \exp \left( -\frac{\sqrt{5}\|\mathbf{z} - \mathbf{z}'\|_2}{\ell} \right).$$

The length-scale parameter  $\ell$  was set to  $\ell = 1.0$ , while the amplitude  $a$  was set to  $a = 9.0$  to ensure that where the posterior distribution is concentrated has significant probability mass under the prior. Results for application of BayesCG to the solution of this PDE, also known as the *forward problem*, are similar to those described in the previous section and are presented in Section S6 of the supplement.

**Inverse Problem** In this section, the solution to the inverse problem when using the BayesCG potential  $\hat{\Phi}$  is compared to the posterior obtained from the exact potential  $\Phi$ . In the latter case CG was used to solve the system to convergence to provide a brute-force benchmark. For BayesCG, the prior was centered,  $\mathbf{x}_0 = \mathbf{0}$ , and the preconditioner prior covariance,  $\Sigma_0 = (P^\top P)^{-1}$ , was used. BayesCG was run to  $m = 80$  iterations, for the mesh with  $N_d = 64$ . This mesh results in a linear system with  $d = 311$ , so 80 iterations represents a relatively small amount of computational effort.

In Figure 5 the posterior distribution over the conductivity field is displayed. In Figures 5b and 5c, respectively, the exact posterior mean and the posterior mean from BayesCG are plotted. Note that, as indicated in the previous section, many of the features of the conductivity field have been recovered even though a relatively small number of iterations have been performed. In Figure 5d the ratio of the pointwise posterior standard deviation from BayesCG to that in CG is plotted. Clearly, throughout the entire spatial domain, the posterior distribution has a larger standard deviation, showing that the posterior uncertainty from BayesCG has successfully been transferred to the posterior over the conductivity field. This results in a posterior distribution which is wider to account for the fact that an imperfect solver was used to solve the forward problem. Overall, the integrated standard deviation over the domain is 0.0365 for BayesCG, while for the exact posterior it is 0.0046.

This example illustrates how BayesCG could be used to relax the computational effort required in EIT in such a way that the posterior is widened to account for the imperfect solution to the forward problem. This setting, as well as other applications of this method, should be explored in more detail in future work.

## 7 Conclusion and Discussion

In this paper we have introduced and theoretically analysed the Bayesian conjugate gradient method, a Bayesian probabilistic numerical method for the solution of linear systems of equations. Given the ubiquity of linear systems in numerical computation,

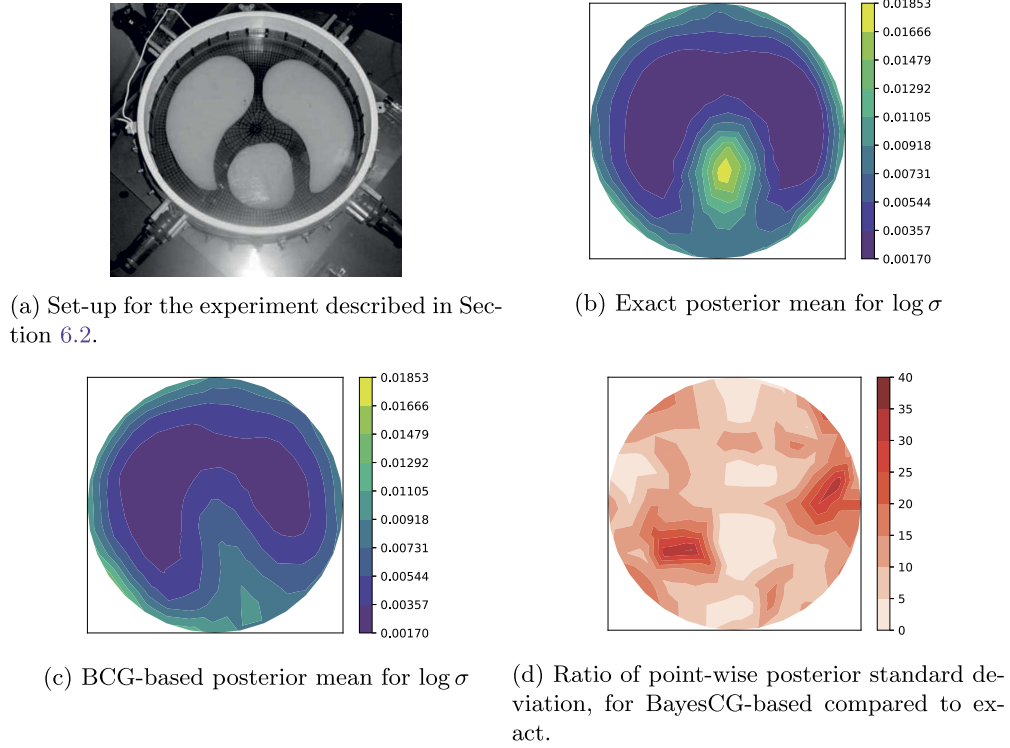


Figure 5: Comparison of the posterior distribution over the conductivity field, when using BayesCG to solve the linear system arising from the forward problem compared to using standard CG.

the question of how to approximate their solution is fundamental. Contrary to CG and other classical iterative methods, BayesCG outputs a probability distribution, providing a principled quantification of uncertainty about the solution after exploring an  $m$ -dimensional subspace of  $\mathbb{R}^d$ . Through the numerical example in Section 6.2 we have shown how this output could be used to make meaningful inferences in applied problems, with reduced computational cost in terms of iterations performed. This could be applied to a broad range of problems in which solution of large linear systems is a bottleneck, examples of which have been given Section 1.1.

**Prior Choice** Prior choice was discussed in detail. An important question that arises here is to what extent the form of the prior can be relaxed. Indeed, in many applied settings information is known about  $\mathbf{x}^*$  which cannot be encoded into a Gaussian prior. For example, the solution of PDEs is often known to be sign-constrained. When encoding this information into the prior it is likely that the conjugacy properties exploited to construct a closed-form posterior will be lost. Then, interrogating such posteriors would require sampling techniques such as the numerical disintegration procedure of Cockayne



et al. (2017), which would incur a dramatically higher cost. Research to determine what prior knowledge can be encoded (either exactly or approximately) without sacrificing numerical performance will be an important future research direction.

It was shown how a numerical analyst’s intuition that the conjugate gradient method “tends to work well” can be encoded into a Krylov-based prior. This went some way towards compensating for the fact that the search directions in BayesCG are constructed in a data-driven manner which is not explicitly acknowledged in the likelihood. Alternative heuristic procedures for calibrating the UQ were explored in the supplement, Section S4.3. An important problem for future research will be to provide practical and theoretically justified methods for ensuring the posterior UQ is well-calibrated.

**Computational Cost and Convergence** The computational cost of BayesCG is only a constant factor higher than that of CG. However, the convergence rates reported in Section 3 can be slower than those of CG. To achieve comparable convergence rates, the prior covariance  $\Sigma_0$  must be chosen to counteract the fact that the rate is based on  $\kappa(\Sigma_0 A^\top A)$  rather than  $\kappa(A)$ , and this can itself incur a substantial computational cost. Future work will focus on reducing the cost associated with BayesCG.

## Supplementary Material

Supplementary Material for “Bayesian Conjugate-Gradient Method”  
(DOI: [10.1214/19-BA1145SUPP](https://doi.org/10.1214/19-BA1145SUPP); .pdf).

## References

- Ajiz, M. A. and Jennings, A. (1984). “A robust incomplete Choleski-conjugate gradient algorithm.” *International Journal for Numerical Methods in Engineering*, 20(5): 949–966. [MR0749826](#). doi: <https://doi.org/10.1002/nme.1620200511>. 2
- Allaire, G. and Kaber, S. M. (2008). *Numerical Linear Algebra*, volume 55 of *Texts in Applied Mathematics*. Springer New York. [MR2365296](#). doi: <https://doi.org/10.1007/978-0-387-68918-0>. 2
- Bartels, S. and Hennig, P. (2016). “Probabilistic Approximate Least-Squares.” In *Proceedings of Artificial Intelligence and Statistics (AISTATS)*. 3
- Benzi, M. (2002). “Preconditioning Techniques for Large Linear Systems: A Survey.” *Journal of Computational Physics*, 182(2): 418–477. [MR1941848](#). doi: <https://doi.org/10.1006/jcph.2002.7176>. 3
- Besag, J. and Green, P. J. (1993). “Spatial statistics and Bayesian computation.” *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 25–37. [MR1210422](#). 2
- Bogachev, V. I. (1998). *Gaussian Measures*, volume 62. American Mathematical Society Providence. [MR1642391](#). doi: <https://doi.org/10.1090/surv/062>. 2

- Bramble, J. H., Pasciak, J. E., and Xu, J. (1990). “Parallel Multilevel Preconditioners.” *Mathematics of Computation*, 55(191): 1–22. MR1023042. doi: <https://doi.org/10.2307/2008789>. 2
- Briol, F.-X., Oates, C. J., Girolami, M., Osborne, M. A., and Sejdinovic, D. (2018). “Probabilistic Integration: A Role in Statistical Computation?” [arXiv:1512.00933](https://arxiv.org/abs/1512.00933). 2
- Calvetti, D., Pitolli, F., Somersalo, E., and Vantaggi, B. (2018). “Bayes Meets Krylov: Statistically Inspired Preconditioners for CGLS.” *SIAM Review*, 60(2): 429–461. MR3797727. doi: <https://doi.org/10.1137/15M1055061>. 3
- Cheng, K.-S., Isaacson, D., Newell, J. C., and Gisser, D. G. (1989). “Electrode models for electric current computed tomography.” *IEEE Transactions on Biomedical Engineering*, 36(9): 918–924. MR1080512. doi: <https://doi.org/10.1137/0150096>. 21
- Cockayne, J., Oates, C. J., Ipsen, I. C. F., and Girolami, M. (2019). “Supplementary Material for “Bayesian Conjugate-Gradient Method”.” *Bayesian Analysis*. doi: <https://doi.org/10.1214/19-BA1145SUPP>. 5
- Cockayne, J., Oates, C., Sullivan, T., and Girolami, M. (2017). “Bayesian Probabilistic Numerical Methods.” [arXiv:1702.03673](https://arxiv.org/abs/1702.03673). 3, 6, 7, 9, 24
- Cockayne, J., Oates, C., Sullivan, T. J., and Girolami, M. (2016). “Probabilistic Meshless Methods for Partial Differential Equations and Bayesian Inverse Problems.” [arXiv:1605.07811v1](https://arxiv.org/abs/1605.07811v1). 2, 12
- Cotter, S. L., Roberts, G. O., Stuart, A. M., and White, D. (2013). “MCMC methods for functions: Modifying old algorithms to make them faster.” *Statistical Science*, 28(3): 424–446. MR3135540. doi: <https://doi.org/10.1214/13-STS421>. 22
- Davis, T. A. (2006). *Direct Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA. MR2270673. doi: <https://doi.org/10.1137/1.9780898718881>. 2
- Diaconis, P. (1988). “Bayesian numerical analysis.” *Statistical Decision Theory and Related Topics IV*, 1: 163–175. MR0927099. 7
- Dunlop, M. M. and Stuart, A. M. (2016). “The Bayesian formulation of EIT: Analysis and algorithms.” *Inverse Problems and Imaging*, 10: 1007–1036. MR3610749. doi: <https://doi.org/10.3934/ipi.2016030>. 21, 22
- Evans, L. (2010). *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*. Providence, Rhode Island: American Mathematical Society, second edition. MR2597943. doi: <https://doi.org/10.1090/gsm/019>. 1, 2
- Fasshauer, G. E. (1999). “Solving differential equations with radial basis functions: Multilevel methods and smoothing.” *Advances in Computational Mathematics*, 11(2–3): 139–159. MR1731694. doi: <https://doi.org/10.1023/A:1018919824891>. 2

- Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). *Bayesian Data Analysis*, volume 2. CRC press Boca Raton, FL. [MR3235677](#).  
13
- Golub, G. H. and Van Loan, C. F. (2013). *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, fourth edition. [MR3024913](#). 12, 14
- Hennig, P. (2015). “Probabilistic Interpretation of Linear Solvers.” *SIAM Journal on Optimization*, 25(1): 234–260. [MR3301314](#). doi: <https://doi.org/10.1137/140955501>. 3, 4, 8, 15
- Hennig, P., Osborne, M. A., and Girolami, M. (2015). “Probabilistic numerics and uncertainty in computations.” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 471(2179): 20150142. [MR3378744](#). doi: <https://doi.org/10.1098/rspa.2015.0142>. 3
- Hestenes, M. R. and Stiefel, E. (1952). “Methods of conjugate gradients for solving linear systems.” *Journal of Research of the National Bureau of Standards*, 49(6): 409. [MR0060307](#). 2, 7
- Holder, D. S. (2004). *Electrical Impedance Tomography: Methods, History and Applications*. CRC Press. 21
- Isaacson, D., Mueller, J. L., Newell, J. C., and Siltanen, S. (2004). “Reconstructions of chest phantoms by the D-bar method for electrical impedance tomography.” *IEEE Transactions on Medical Imaging*, 23(7): 821–828. 22
- Larkin, F. M. (1972). “Gaussian measure in Hilbert space and applications in numerical analysis.” *The Rocky Mountain Journal of Mathematics*, 2(3): 379–421. [MR0303193](#). doi: <https://doi.org/10.1216/RMJ-1972-2-3-379>. 2, 3
- Liesen, J. and Strakos, Z. (2012). *Krylov Subspace Methods*. Principles and Analysis. Oxford University Press. [MR3024841](#). 2, 11
- Oates, C. J., Cockayne, J., Aykroyd, R. G., and Girolami, M. (2019). “Bayesian Probabilistic Numerical Methods in Time-Dependent State Estimation for Industrial Hydrocyclone Equipment.” *Journal of the American Statistical Association*. To appear. 21
- Owhadi, H. (2015). “Bayesian numerical homogenization.” *Multiscale Modeling & Simulation*, 13(3): 812–828. [MR3369060](#). doi: <https://doi.org/10.1137/140974596>. 12
- Parker, A. and Fox, C. (2012). “Sampling Gaussian distributions in Krylov spaces with conjugate gradients.” *SIAM Journal on Scientific Computing*, 34(3): B312–B334. [MR2970281](#). doi: <https://doi.org/10.1137/110831404>. 2
- Rasmussen, C. E. (2004). “Gaussian Processes in Machine Learning.” In *Advances in Intelligent Data Analysis VIII*, 63–71. Berlin, Heidelberg: Springer Berlin Heidelberg. 2

- Reinarz, A., Dodwell, T., Fletcher, T., Seelinger, L., Butler, R., and Scheichl, R. (2018). “Dune-composites – A new framework for high-performance finite element modelling of laminates.” *Composite Structures*, 184: 269–278. 2
- Roeckner, E., Bäuml, G., Bonaventura, L., Brokopf, R., Esch, M., and Giorgetta, M. (2003). “The atmospheric general circulation model ECHAM 5. PART I: Model description.” Technical report, MPI für Meteorologie. 3
- Saad, Y. (1994). “ILUT: A dual threshold incomplete LU factorization.” *Numerical Linear Algebra with Applications*, 1(4): 387–402. MR1306700. doi: <https://doi.org/10.1002/nla.1680010405>. 2
- Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition. MR1990645. doi: <https://doi.org/10.1137/1.9780898718003>. 2, 3
- Schäfer, F., Sullivan, T. J., and Owhadi, H. (2017). “Compression, inversion, and approximate PCA of dense kernel matrices at near-linear computational complexity.” [arXiv:1706.02205](https://arxiv.org/abs/1706.02205). 2
- Shewchuk, J. R. (1994). “An introduction to the conjugate gradient method without the agonizing pain.” Technical report. 29
- Somersalo, E., Cheney, M., and Isaacson, D. (1992). “Existence and uniqueness for electrode models for electric current computed tomography.” *SIAM Journal on Applied Mathematics*, 52(4): 1023–1040. MR1174044. doi: <https://doi.org/10.1137/0152060>. 21
- Stuart, A. M. (2010). “Inverse problems: A Bayesian perspective.” *Acta Numerica*, 19: 451–559. MR2652785. doi: <https://doi.org/10.1017/S0962492910000061>. 5
- Tikhonov, A. N. (1963). “On the solution of ill-posed problems and the method of regularization.” In *Doklady Akademii Nauk*, volume 151, 501–504. Russian Academy of Sciences. MR0162377. 5
- Traub, J. F., Wasilkowski, G. W., and Woźniakowski, H. (1988). *Information-Based Complexity*. Computer Science and Scientific Computing. Academic Press, Inc., Boston, MA. With contributions by A. G. Werschulz and T. Boulton. MR0958691. 5, 8
- Wikle, C. K., Milliff, R. F., Nychka, D., and Berliner, L. M. (2001). “Spatiotemporal Hierarchical Bayesian Modeling Tropical Ocean Surface Winds.” *Journal of the American Statistical Association*, 96(454): 382–397. MR1939342. doi: <https://doi.org/10.1198/016214501753168109>. 2

### Acknowledgments

Chris J. Oates and Mark Girolami were supported by the Lloyd’s Register Foundation programme on Data-Centric Engineering. Ilse C.F. Ipsen was supported in part by NSF grant DMS-1760374. Mark Girolami was supported by EPSRC grants [EP/R034710/1, EP/R018413/1, EP/R004889/1, EP/P020720/1], an EPSRC Established Career Fellowship [EP/J016934/3] and a Royal Academy of Engineering Research Chair.

This material was based upon work partially supported by the National Science Foundation under Grant DMS-1127914 to the Statistical and Applied Mathematical Sciences Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

The authors are grateful to Shewchuk ([1994](#)). Techniques described therein formed the basis of some of the proofs in this paper. They would also like to thank Philipp Hennig and Simon Bartels for their help in the implementation of their algorithm and Tim Sullivan for his feedback on the manuscript.